EUROPEAN OFFICE OF AEROSPACE RESEARCH AND DEVELOPMENT

Contract F61775-00-WE068

12 September, 2000

PROGRESS REPORT,

17 April, 2001

NEURAL NETWORKS CONTROL OF A MAGNETIC LEVITATION SYSTEM

Principal Investigator: Prof. Chaïban NASR

Lebanese University - Faculty of Engineering, Section I
Dept. of Electrical and Electronic Engineering
P.O. Box N° 7 - Zgharta - North Lebanon
LEBANON

Phone: (961) 3 369245
Fax: (961) 6 385089
E-mail: chnasr@ul.edu.lb

1-INTRODUCTION:

Perhaps the most innovative technical development over the last ten years in the field of control has been the introduction of artificial neural networks (ANN) methods for identification, modeling and control [1,2]. The basic concept of artificial neural nets stems from the idea of modeling individual brain cells/neurons in a fairly simple way, and then connecting these models in a highly parallel fashion to offer a complex processing mechanism which exhibits learning in terms of its overall nonlinear characteristics. Actual brain cells are not of one particular form. They are not all identical, whereas at present artificial neural networks tend to consist of one type of neuron. Further, the overall make up of a brain, in terms of connectivity and structure, is highly complex and not well understood, whereas artificial neural networks are generally well structured and simply coupled, thereby enabling the possibility of understanding their mode of operation.

In terms of a control systems environment, the majority of practical controllers actually in use are both simple and linear, and are directed towards the control of a plant which is either reasonably linear or at least linearisable. However, it is sufficient to state that a neural network is usually a complex

Form SF298 Citation Data

Report Date ("DD MON YYYY")	Report Type	Dates Covered (from... to) ("DD MON YYYY")
20042001	Final	

Title and Subtitle Neural Networks Control of a Magnetic Levitation System	**Contract or Grant Number** F61775-00-WE068
	Program Element Number
Authors Nasr, Chaiban	**Project Number**
	Task Number
	Work Unit Number
Performing Organization Name(s) and Address(es) Lebanese University-Faculty of Engineering, Section I El-Kobbeh Tripoli North Lebanon	**Performing Organization Number(s)**
Sponsoring/Monitoring Agency Name(s) and Address(es) EOARD PSC 802 BOX 14 FPO 09499-0200	**Monitoring Agency Acronym**
	Monitoring Agency Report Number(s)

Distribution/Availability Statement
Approved for public release, distribution unlimited

Supplementary Notes

Abstract
This report results from a contract tasking Lebanese University-Faculty of Engineering, Section I as follows: The purpose of the work is a simulating investigation of the use of artificial neural networks (ANN) in conjunction of proportional-integral-derivative (PID) controllers in control of non-contacting active magnetic bearings (AMB). The objective of this technique is to reduce the effect of the unbalance on the rotor displacement without the estimating perturbation. The work consists of the following: 1) application of artificial neural networks (multi-layer perceptrons) for nonlinear model of the active magnetic bearing by using the dynamic back-propagation methods for the adjustment of parameters; and 2) application of artificial neural networks in controlling closed-loop active magnetic bearing and comparison with the use of PID controllers. The obtained results should create a basis for a further research program connecting the fundamental knowledge with practical applications.

Subject Terms
EOARD; Control; Neural Networks

Document Classification unclassified	**Classification of SF298** unclassified

Classification of Abstract unclassified	**Limitation of Abstract** unlimited
Number of Pages 21	

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	20-April 2001	Final Report

4. TITLE AND SUBTITLE

Neural Networks Control of a Magnetic Levitation System

5. FUNDING NUMBERS

F61775-00-WE068

6. AUTHOR(S)

Professor Chaiban Nasr

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Lebanese University-Faculty of Engineering, Section I
El-Kobbeh
Tripoli North
Lebanon

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

EOARD
PSC 802 BOX 14
FPO 09499-0200

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

SPC 00-4068

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

A

13. ABSTRACT (Maximum 200 words)

This report results from a contract tasking Lebanese University-Faculty of Engineering, Section I as follows: The purpose of the work is a simulating investigation of the use of artificial neural networks (ANN) in conjunction of proportional-integral-derivative (PID) controllers in control of non-contacting active magnetic bearings (AMB). The objective of this technique is to reduce the effect of the unbalance on the rotor displacement without the estimating perturbation. The work consists of the following: 1) application of artificial neural networks (multi-layer perceptrons) for nonlinear model of the active magnetic bearing by using the dynamic back-propagation methods for the adjustment of parameters; and 2) application of artificial neural networks in controlling closed-loop active magnetic bearing and comparison with the use of PID controllers. The obtained results should create a basis for a further research program connecting the fundamental knowledge with practical applications.

14. SUBJECT TERMS

EOARD, Control, Neural Networks

15. NUMBER OF PAGES

21

16. PRICE CODE

N/A

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19, SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

nonlinear mapping tool, and the use of such a device on relatively simple linear problems makes very little sense at all, being a case of over-skill [3]. The fact that neural networks have the capability of dealing with complex non-linearity in a fairly general way is an exciting feature. By their nature, nonlinear systems are non-uniform and invariably require custom designed control schemes to deal with individual characteristics. No general theory deals comprehensively with the wide range of nonlinear systems encountered, so an approach, namely, neural networks that can offer a fairly broad coverage are therefore extremely attractive. By viewing neural networks with control applications clearly in mind, a number of observations can be made [4], the first of these being perhaps the most significant:

(i) Neural networks can flexibly and arbitrarily map nonlinear functions. Such networks are best suited for the control of nonlinear systems.
(ii) Neural networks are particularly well suited to multivariable applications due to their ability to map interactions and cross-couplings readily whilst incorporating many inputs and outputs.
(iii) Networks can either be trained off-line and subsequently employed either on or off-line, or they can be trained on-line as part of an adaptive control scheme or simply a real-time system identifier. Understanding of a network's mode of operation within an adaptive controller is, at the present time, extremely limited.
(iv) Neural networks are inherently parallel processing devices which exhibit to an extent, at least, fault tolerant characteristics.

The purpose of the work is a simulating investigation of the use of artificial neural networks (ANN) in conjunction of PID controllers in control of non-contacting active magnetic bearings (AMB). The objective of this technique is to reduce the effect of the unbalance on the rotor displacement without the estimating perturbation. Another purpose will be the application of this knowledge for the conception of different types of architectures of the ANN in closed-loop systems.

2 - ACTIVE MAGNETIC BEARING (AMB):

2-1 - Background:

Active magnetic bearing are used to support radial and thrust loads in precision rotating machinery or to levitate linear motion devices. Since there can be no mechanical contact with active magnetic bearing, the speed of supported components is not limited. Due to no friction with active magnetic bearing, only the actuators, sensors, and servo system used limit motion resolution of the supported object. Thus active magnetic bearings can be used in virtually any environment as long as the electromagnetic coils are well protected, e.g., they can be operated in the air with temperatures ranging from -235^{o}C to 450^{o}C [5]. Active magnetic bearing actuators can be used for improving the dynamic behavior of the supported load, however, they are open loop unstable and require a closed-loop control system for stability [5], [6]. As active magnetic bearings are used to support rotors, the three dimensional rotor dynamics and the electromagnetic subsystems are severely coupled, and the whole active magnetic bearing system is highly nonlinear and complex. The problems of control design for an active magnetic bearing are thus challenging and play a key role in the design of machines with nanometer accuracy.

Jackson and Destombes [7] considered the usual efficient type of active magnetic bearings in the attraction mode. Salm and Schweizer [8] considered the modeling and control of active magnetic bearings with a flexible rotor. The flexibility effect due to eccentricity between the center of mass and geometric center is critical at high operational speeds. Matsumura and Yashimoto [9] considered the modeling and control of a horizontal shaft with active magnetic bearings. They suggested a linear quadratic regulation (LQR) control with incorporated integral control. Matsumura et al. [10] proposed an axial thrust and radial control design methodology, and experimental results are shown to validate the effectiveness of the control strategy under wide operation ranges. Youcef-Toumi and Reddy [11] proposed a model with actuator dynamics and rotor flexibility for active magnetic bearings, and experimental results using time delay control show that the suggested approach can obtain good disturbance rejection. The above approaches to the control of magnetic bearings are all based on the linear approximation models, thus the gyroscopic effects among the rotational degrees of freedom and the mutual induction between the electromagnets are not fully considered in the control design.

2-2 - Modeling of an Active Magnetic Bearing:

The device used for simulations is a dynamic modeling of a rigid rotor. This rotor is suspended against gravity by an active magnetic bearing and centered by two active magnetic bearings. Active magnetic bearings with air gaps are used for generating the required radial and axial control force components to fully regulating the floating rotor as shown in figure 1.

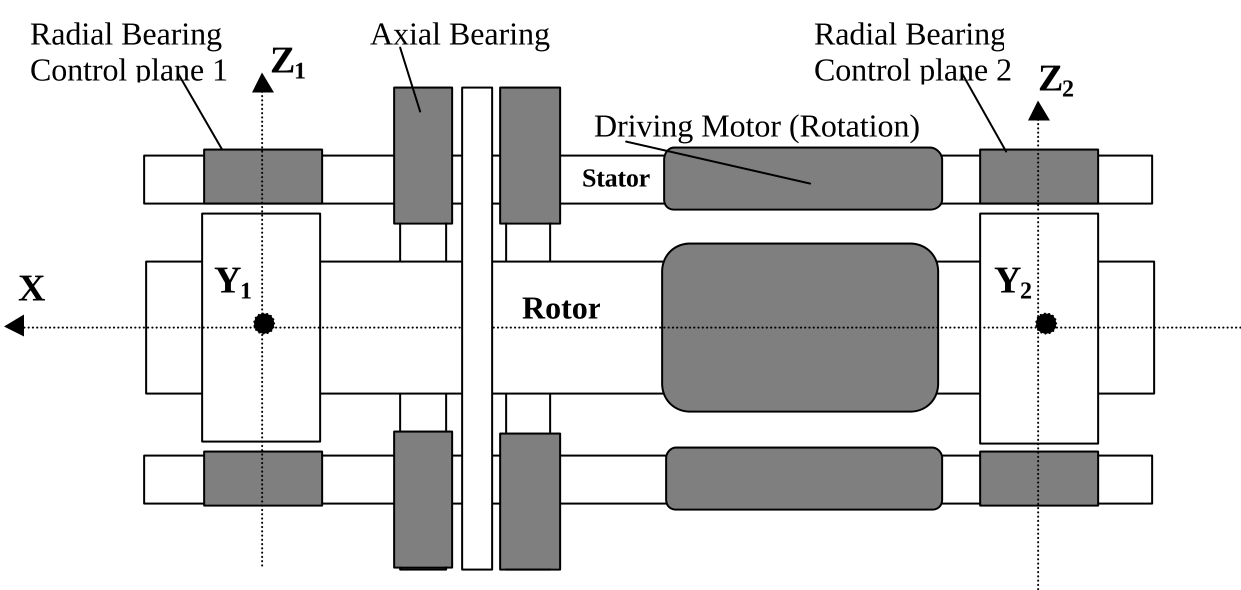


Fig. 1: - Schematic representation of a rotor with Active Magnetic Bearings.

A. Five Mechanical Degree-of-Freedom:

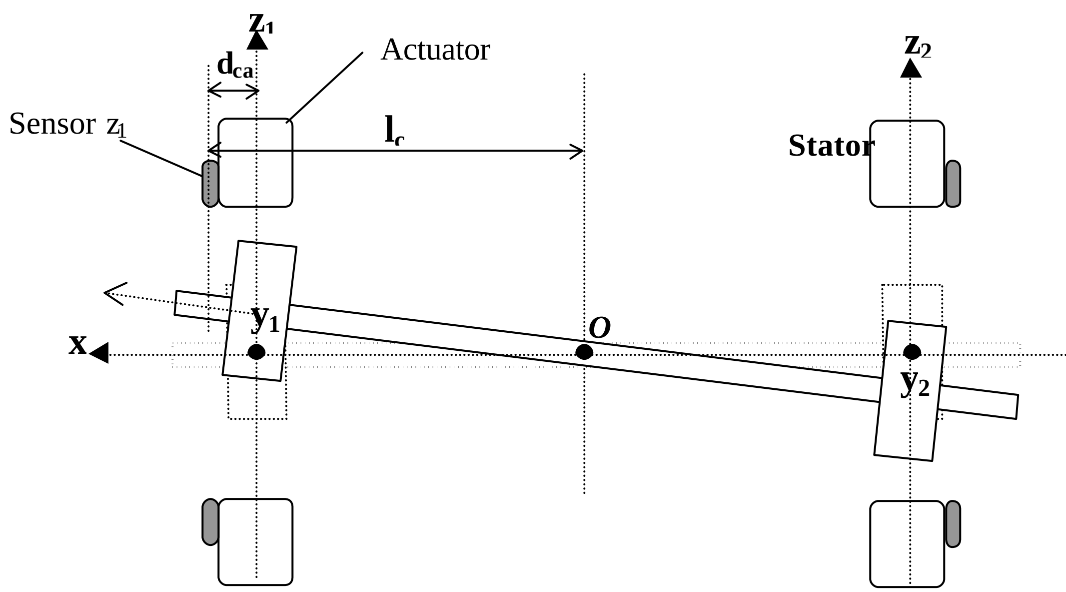


Fig. 2: - Schematic representation of a rotor in the (x, z) plane.

Consider the simplified model of an active magnetic bearing system shown in figure 2. It consists of two planes of control with one active axial bearing. For simplicity, the rotor is assumed rigid and its center of mass and geometric center are consistent, i.e., with no eccentricity. The five mechanical Degree-Of-Freedom considered will be measured by the sensors, they are:

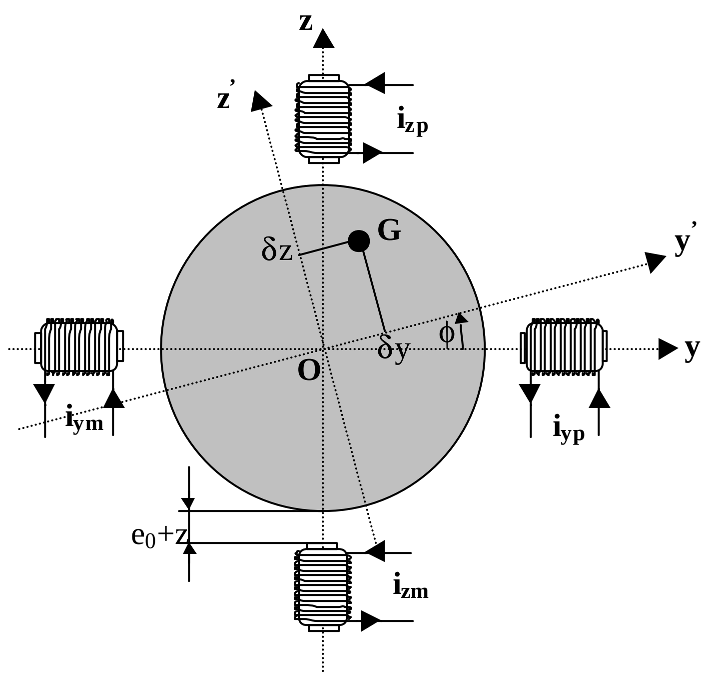
- x which represents the geometric center *O* of the rotor.
- y_1 and z_1 (respectively y_2 and z_2) measured in the control plane 1 (respectively in the control plane 2).

The parameter l_c represents the distance between the radial position sensor and the geometric center. The parameter d_{ca} represents the distance between the sensor and the corresponding actuator.
Another Degree-Of-Freedom may be the rotation angle in the x direction. But this variable can be controlled independently.

B. Modeling of the (y, z) plane control:

A plane system is the representation of a mobile in the form of a disc where the translation position is controlled by four actuators placed in the plane control described in the figure 3. This mobile cannot rotate around the y- and z-axis; it can only rotate around the x-axis with a certain velocity.

The coordinates of the center *O* in the fixed reference frame attached to the stator are (x, y, z), and the coordinates of the geometric center G in the moving frame attached to the rotor (y', z') are (0, δy, δz). This moving frame rotate with an angle ϕ with respect to the frame (y, z) and parallel to the fixed frame attached to the stator.



<u>Fig. 3:</u> - Schematic representation of the plane control (y, z).

By applying the Lagrange's equations, the dynamic equation is represented as follow:

$$D(q)\cdot\ddot{q}+C(q,\dot{q})\cdot\dot{q}+\frac{\partial\Re}{\partial\dot{q}}=B\cdot F$$

Where D, C, and B are matrices, and F is the input matrix.

The mechanical energy of the system is given by the equation:

$$T_m=\frac{1}{2}\left(m\left|V_G\right|^2+I_x\dot{f}^2\right)$$

And the electrical energy of the system is given by the equation:

$$T_e=\frac{1}{2}\left(\frac{l\,i_{yp}^2}{e_0-y}+\frac{l\,i_{ym}^2}{e_0+y}+\frac{l\,i_{zp}^2}{e_0-z}+\frac{l\,i_{zm}^2}{e_0+z}\right)$$

By recombining the expressions of the mechanical and electrical energy and by considering δz=0, we obtain:

For T_m :

$$T_m=\frac{1}{2}\dot{q}_m^T\cdot D_m\cdot\dot{q}_m$$

$$où:\dot{q}_m=\left[\dot{x},\dot{y},\dot{z},\dot{f}\right]^T$$

$$et:D_m=\begin{bmatrix} m & 0 & 0 & 0 \\ 0 & m & 0 & -mdy\sin(f) \\ 0 & 0 & m & mdy\cos(f) \\ 0 & -mdy\sin(f) & mdy\cos(f) & mdy^2+I_x \end{bmatrix}$$

and for T_e :

$$
\begin{aligned}
&T_e = \frac{1}{2}\dot{q}_e^T \cdot D_e \cdot \dot{q}_e \\
&où : \dot{q}_e = \left[i_{yp}, i_{ym}, i_{zp}, i_{zm}\right]^T \\
&et : D_e = \begin{bmatrix} \frac{l}{e_0 - y} & 0 & 0 & 0 \\ 0 & \frac{l}{e_0 + y} & 0 & 0 \\ 0 & 0 & \frac{l}{e_0 - z} & 0 \\ 0 & 0 & 0 & \frac{l}{e_0 + z} \end{bmatrix}
\end{aligned}
$$

The expression of the total energy will be condensed as follows:

$$
\begin{aligned}
&T = \frac{1}{2}\dot{q}^T \cdot D \cdot \dot{q} \\
&où : \dot{q} = \left[\dot{q}_m, \dot{q}_e\right]^T \\
&et : D = \begin{bmatrix} D_m & 0 \\ 0 & D_e \end{bmatrix}
\end{aligned}
$$

On the other hand, the expression of the matrix C is given by the relation [12]:

$$
C = \begin{bmatrix}
0 & 0 & -mdy\cos(f)\dot{f} & \frac{1}{2}\frac{-l\,i_{yp}}{(e_0-y)^2} & \frac{1}{2}\frac{l\,i_{ym}}{(e_0+y)^2} & 0 & 0 \\
0 & 0 & -mdy\sin(f)\dot{f} & 0 & 0 & \frac{1}{2}\frac{-l\,i_{zp}}{(e_0-z)^2} & \frac{1}{2}\frac{l\,i_{zm}}{(e_0+z)^2} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2}\frac{-l\,i_{yp}}{(e_0-y)^2} & 0 & 0 & \frac{1}{2}\frac{l\dot{y}}{(e_0-y)^2} & 0 & 0 & 0 \\
\frac{1}{2}\frac{-l\,i_{ym}}{(e_0+y)^2} & 0 & 0 & 0 & \frac{1}{2}\frac{-l\dot{y}}{(e_0+y)^2} & 0 & 0 \\
0 & \frac{1}{2}\frac{l\,i_{zp}}{(e_0+z)^2} & 0 & 0 & 0 & \frac{1}{2}\frac{l\dot{z}}{(e_0-z)^2} & 0 \\
0 & \frac{1}{2}\frac{-l\,i_{zm}}{(e_0+z)^2} & 0 & 0 & 0 & 0 & \frac{1}{2}\frac{-l\dot{z}}{(e_0+z)^2}
\end{bmatrix}
$$

We define P_y and P_z the gravity forces. By defining E_{yp}, E_{ym}, E_{zp}, E_{zm} the applied voltages for the four actuators, the global system can be modeled by:

$$
\begin{cases}
m\ddot{y} = mdy\sin(f)\ddot{f} + mdy\cos(f)(\dot{f})^2 + \dfrac{1}{2}\dfrac{l\,i_{yp}^2}{(e_0 - y)^2} - \dfrac{1}{2}\dfrac{l\,i_{ym}^2}{(e_0 + y)^2} + P_y \\
m\ddot{z} = -mdy\cos(f)\ddot{f} + mdy\sin(f)(\dot{f})^2 + \dfrac{1}{2}\dfrac{l\,i_{zp}^2}{(e_0 - z)^2} - \dfrac{1}{2}\dfrac{l\,i_{zm}^2}{(e_0 + z)^2} + P_z \\
(mdy^2 + I_x)\ddot{f} = mdy\sin(f)\ddot{y} - mdy\cos(f)\ddot{z} \\
\dfrac{l\,\dot{i}_{yp}}{e_0 - y} = -\dfrac{l\,i_{yp}\dot{y}}{(e_0 - y)^2} - Ri_{yp} + E_{yp} \\
\dfrac{l\,\dot{i}_{ym}}{e_0 + y} = \dfrac{l\,i_{ym}\dot{y}}{(e_0 + y)^2} - Ri_{ym} + E_{ym} \\
\dfrac{l\,\dot{i}_{zp}}{e_0 - z} = -\dfrac{l\,i_{zp}\dot{z}}{(e_0 - z)^2} - Ri_{zp} + E_{zp} \\
\dfrac{l\,\dot{i}_{zm}}{e_0 + z} = \dfrac{l\,i_{zm}\dot{z}}{(e_0 + z)^2} - Ri_{zm} + E_{zm}
\end{cases}
$$

C. Modeling the five Degree-Of-Freedom:

By assuming that the rotor is rigid and its center of mass and geometric center are consistent, we have $\delta x = \delta y = \delta z = 0$. We consider also that the system is symmetric in the y- and z-plane, we have $I_y = I_z$. We considered also that the distance d_{ca} is equal to zero (the distance between the actuator and the associated sensor is equal to zero).

Two types of equations will represent the simulated model:

- Six mechanical equations relating the expressions of the five positions (x, y_1, z_1, y_2, z_2) and of the angle of rotation ϕ of the system to the different parameters of the system.
- Ten electrical equations relating the expressions of the ten currents (i_{xp}, i_{xm}, i_{y1p}, i_{y1m}, i_{z1p}, i_{z1m}, i_{y2p}, i_{y2m}, i_{z2p}, i_{z2m}) to the expressions of the mechanical forces applied to the system. For every position, we have two actuators: one in the positive direction (suffix p) and the other in the negative direction (suffix m).

2-3 - Control of the Active Magnetic Bearing:

Before introducing the neural network controllers, we have to simulate a control technique based on linearized PID controllers. This phase is important to study the effect of using new techniques based on neural networks controllers.

The proposed control technique is based on a cascade structure as shown in figure 4. We begin by deciding of the kinematics of the system followed by the dynamic equations that will determine the different forces applied. From the forces applied, we determine the expressions of the currents that must control the actuators.

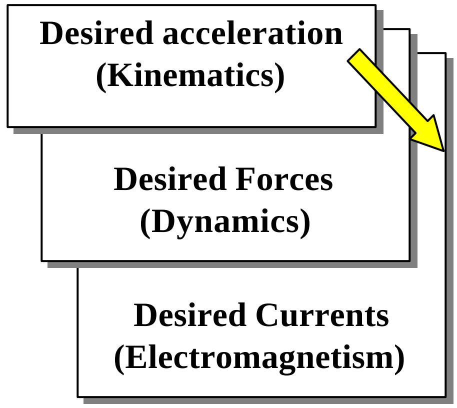


Fig. 4: - Cascade Structure of the control technique.

The equations relating the expressions of the desired forces to the expressions of the desired currents are given by:

$$\frac{1}{2}\frac{l\,i_{xp}^2}{(-e_0+x)^2}-\frac{1}{2}\frac{l\,i_{xm}^2}{(e_0+x)^2}=F_{ix}$$

$$\frac{l\,i_{y1p}^2}{(-2e_0l_c+2y_1l_c-d_{ca}y_1+d_{ca}y_2)^2}-\frac{l\,i_{y1m}^2}{(2e_0l_c+2y_1l_c-d_{ca}y_1+d_{ca}y_2)^2}=F_{iy1}$$

$$\frac{l\,i_{z1p}^2}{(-2e_0l_c+2z_1l_c-d_{ca}z_1+d_{ca}z_2)^2}-\frac{l\,i_{z1m}^2}{(2e_0l_c+2z_1l_c-d_{ca}z_1+d_{ca}z_2)^2}=F_{iz1}$$

$$\frac{l\,i_{y2p}^2}{(-2e_0l_c+2y_2l_c+d_{ca}y_1-d_{ca}y_2)^2}-\frac{l\,i_{y2m}^2}{(2e_0l_c+2y_2l_c+d_{ca}y_1-d_{ca}y_2)^2}=F_{iy2}$$

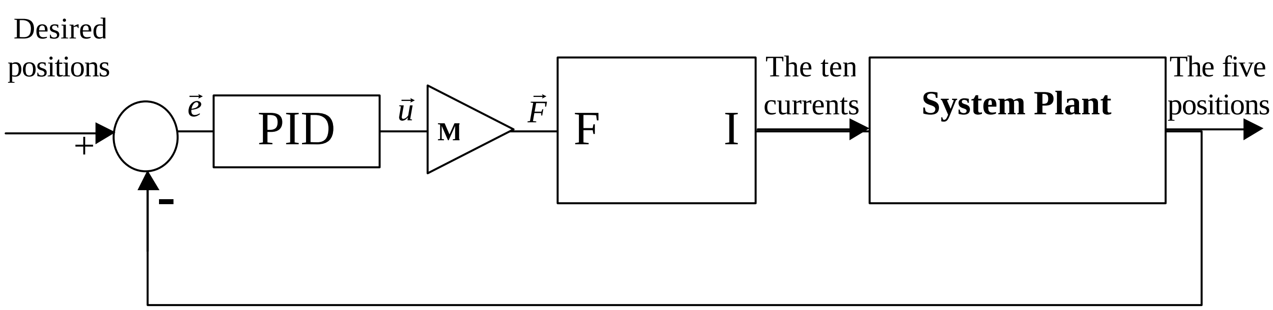
$$\frac{l\,i_{z2p}^2}{(-2e_0l_c+2z_2l_c+d_{ca}z_1-d_{ca}z_2)^2}-\frac{l\,i_{z2m}^2}{(2e_0l_c+2z_2l_c+d_{ca}z_1-d_{ca}z_2)^2}=F_{iz2}$$

We recognize from this system of equations that every relation consists of two unknown currents of the same axe of control. For that, we suppose that one current is fixed and the other is to be calculated. If the desired force is positive (respectively negative), the current related to the actuator that create a negative (respectively positive) acceleration will be fixed as a parameter. The commutation of the current is perfect, which means that only one actuator is functioning in a given direction. This will lead to the fact that, in every direction, the current fixed as a parameter will be equal to zero.

On the other hand, the electromechanical equations representing the five positions and the velocity of the rigid rotor in function of the different currents and of the dynamic of the system are given by:

$$
\begin{bmatrix} \ddot{x} \\ \ddot{y}_1 \\ \ddot{z}_1 \\ \ddot{y}_2 \\ \ddot{z}_2 \\ \ddot{f} \end{bmatrix}
=
\begin{bmatrix}
\frac{1}{m}\left(\frac{1}{2}\cdot\frac{l_x i_{xp}^2}{\left(-e_0+x\right)^2}-\frac{1}{2}\cdot\frac{l_x i_{xm}^2}{\left(e_0+x\right)^2}\right) \\
\frac{\left(I_y+ml_c^2\right)}{I_y m}\left(-\frac{1}{4}\cdot\frac{I_x\dot{z}_1\dot{f}}{l_c^2}+\frac{1}{4}\cdot\frac{I_x\dot{z}_2\dot{f}}{l_c^2}+\frac{1}{2}\cdot\frac{l i_{y1p}^2}{\left(e_0-y_1\right)^2}-\frac{1}{2}\cdot\frac{l i_{y1m}^2}{\left(e_0+y_1\right)^2}\right) \\
+\frac{\left(-I_y+ml_c^2\right)}{I_y m}\left(-\frac{1}{4}\cdot\frac{I_x\dot{z}_1\dot{f}}{l_c^2}+\frac{1}{4}\cdot\frac{I_x\dot{z}_2\dot{f}}{l_c^2}-\frac{1}{2}\cdot\frac{l i_{y2p}^2}{\left(e_0-y_2\right)^2}+\frac{1}{2}\cdot\frac{l i_{y2m}^2}{\left(e_0+y_2\right)^2}\right) \\
\frac{\left(I_y+ml_c^2\right)}{I_y m}\left(\frac{1}{4}\cdot\frac{I_x\dot{y}_1\dot{f}}{l_c^2}-\frac{1}{4}\cdot\frac{I_x\dot{y}_2\dot{f}}{l_c^2}+\frac{1}{2}\cdot\frac{l i_{z1p}^2}{\left(e_0-z_1\right)^2}-\frac{1}{2}\cdot\frac{l i_{z1m}^2}{\left(e_0+z_1\right)^2}\right) \\
+\frac{\left(-I_y+ml_c^2\right)}{I_y m}\left(\frac{1}{4}\cdot\frac{I_x\dot{y}_1\dot{f}}{l_c^2}-\frac{1}{4}\cdot\frac{I_x\dot{y}_2\dot{f}}{l_c^2}-\frac{1}{2}\cdot\frac{l i_{z2p}^2}{\left(e_0-z_2\right)^2}+\frac{1}{2}\cdot\frac{l i_{z2m}^2}{\left(e_0+z_2\right)^2}\right) \\
\frac{\left(-I_y+ml_c^2\right)}{I_y m}\left(\frac{1}{4}\cdot\frac{I_x\dot{z}_1\dot{f}}{l_c^2}-\frac{1}{4}\cdot\frac{I_x\dot{z}_2\dot{f}}{l_c^2}-\frac{1}{2}\cdot\frac{l i_{y1p}^2}{\left(e_0-y_1\right)^2}+\frac{1}{2}\cdot\frac{l i_{y1m}^2}{\left(e_0+y_1\right)^2}\right) \\
-\frac{\left(I_y+ml_c^2\right)}{I_y m}\left(-\frac{1}{4}\cdot\frac{I_x\dot{z}_1\dot{f}}{l_c^2}+\frac{1}{4}\cdot\frac{I_x\dot{z}_2\dot{f}}{l_c^2}-\frac{1}{2}\cdot\frac{l i_{y2p}^2}{\left(e_0-y_2\right)^2}+\frac{1}{2}\cdot\frac{l i_{y2m}^2}{\left(e_0+y_2\right)^2}\right) \\
\frac{\left(-I_y+ml_c^2\right)}{I_y m}\left(-\frac{1}{4}\cdot\frac{I_x\dot{y}_1\dot{f}}{l_c^2}+\frac{1}{4}\cdot\frac{I_x\dot{y}_2\dot{f}}{l_c^2}-\frac{1}{2}\cdot\frac{l i_{z1p}^2}{\left(e_0-z_1\right)^2}+\frac{1}{2}\cdot\frac{l i_{z1m}^2}{\left(e_0+z_1\right)^2}\right) \\
-\frac{\left(I_y+ml_c^2\right)}{I_y m}\left(\frac{1}{4}\cdot\frac{I_x\dot{y}_1\dot{f}}{l_c^2}-\frac{1}{4}\cdot\frac{I_x\dot{y}_2\dot{f}}{l_c^2}-\frac{1}{2}\cdot\frac{l i_{z2p}^2}{\left(e_0-z_2\right)^2}+\frac{1}{2}\cdot\frac{l i_{z2m}^2}{\left(e_0+z_2\right)^2}\right) \\
\frac{1}{I_x}\left(\frac{1}{4}\cdot\frac{I_x\dot{y}_1\dot{z}_1}{l_c^2}-\frac{1}{4}\cdot\frac{I_x\dot{y}_1\dot{z}_2}{l_c^2}-\frac{1}{4}\cdot\frac{I_x\dot{y}_2\dot{z}_1}{l_c^2}+\frac{1}{4}\cdot\frac{I_x\dot{y}_2\dot{z}_2}{l_c^2}\right)
\end{bmatrix}
$$

These equations (positions and currents) will lead to a closed loop system represented in figure 5.



<u>Fig. 5:</u> - Schematic representation of the linearized control technique.

<u>3 - MULTI-LAYER PERCEPTRONS METHOD (MLP):</u>

<u>3-1 - Introduction:</u>

Multi-layer perceptrons [13] provide one arrangement for neural network implementation, by means of nonlinear relationships between, firstly, the network inputs to outputs and, secondly, the network parameters to outputs.

Such a network consists of a number of neuron layers, n, linking its input vector, u, to its output vector, y, by means of the equation:

$$y = \varphi_n(W_n\varphi_{n-1}(W_{n-1}\ldots\varphi_1(W_1 u + b_1) + \ldots + b_{n-1}) + b_n)$$

in which W_i is the weight matrix associated with the layer, φ_i is a nonlinear operator associated with the ith layer and b_i indicate threshold or bias values associated with each node in the ith layer. The function φ_i is a sigmoid function for all n.

It is known in reality that real neurons, located in different areas of the nervous system, have different modes of behavior [14] ranging from Gaussian-like for visual needs to sigmoid for ocular motor needs. It is generally the case for artificial neural networks, however, that only one type of non-linearity is employed for a particular network, this linking in closely with the fact that each network is only employed for one particular task.

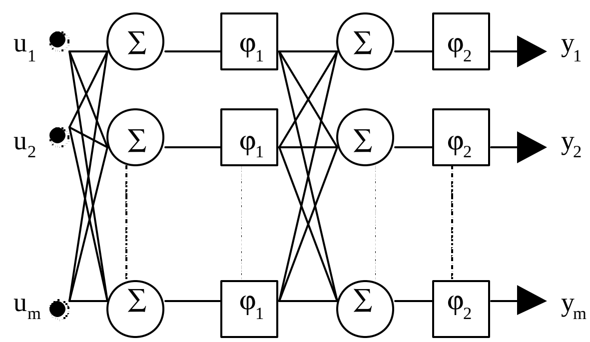


Fig.6: - A two-layer multi-layer perceptron.

Figure 6 shows a fully connected network in that all of the neuron outputs in one layer of the network are connected as inputs to the next layer. This is normal practice: however, it is quite possible for part-connectivity to be realized by connecting a group of outputs to only specific input. By this means sub-models can be formed within the overall MLP, and these can be particularly useful where a specific system characteristic is to be dealt with or inputs/outputs from the system can be categorized into certain types.

Whatever the network connectivity, key questions in the use of MLPs are how many layers there should be and how many neurons there should be in each layer. Once these structural features of the network have been selected, it remains for the adjustable weights to be settled on such that the network is completely specified in terms of its functionality.

3-2 - Dynamic back propagation learning:

By far the most popular method employed for weight training in MLP neural networks is called back propagation [15]. In the standard feed-forward MLP network, back propagation solves the problem of missing information to the hidden layers, neither the input to nor the reference signals for the hidden layers are known. This solving problem is obtained by taking the inputs to the hidden layers as being the inputs to the first layers propagated through the network. The reference signals for the hidden layers are then obtained by error back propagation through the network. This is realized by obtaining the partial derivative of the squared error with respect to the parameters.

It is worth pointing out that the back propagation algorithm has also been used for weight learning in feedback neural networks [16,17], these being networks in which the network structure incorporates feedback, whereby the output of every neuron is fed back, in weighted form, to the input of every neuron. The architecture of such a network is inherently dynamic and realizes powerful capabilities due to its complexity.

Consider, as a starting point, a single neuron with output y_i; then

$$y_i = \varphi(x_i) = \frac{1 - \exp(-x_i)}{1 + \exp(-x_i)}$$

In which:

$$x_i = \sum_{i=1}^{m} w_{ij} u_j + w_0$$

In this expression w_0 is a bias term. If it is assumed that at an instant in time, for an input u_j the output y_i should be equal to the desired output y_d, then the squared error of the output signal is given by:

$$E_i = \frac{1}{2}(y_d - y_i)^2 = \frac{1}{2} e_i^2$$

and it is desired to minimize E_i by means of a suitable/best choice of the weighting coefficients w_{ij} .

Consider the problem of minimizing the scalar error function E(W), where W is a vector of weights to be adjusted by means of an interactive procedure generating a number of search points, W(k), such that:

$$W(k+1) = \alpha(k)W(k) + \eta(k)d(k)$$

In this relation an initial set of weightings W(0) is made through prior knowledge, by a reasoned guess or even relatively randomly. The term $\alpha(k)W(k)$ represent a momentum term and $\alpha(k)$ is usually a positive number called the momentum constant. The term d(k) indicates the search direction, whereas $\eta(k)$ indicates the length of search step or the amount of learning to be carried out.

In this way, the weights associated with one neuron can be adjusted in order to minimize the squared error. The approach can then be extended in order to adjust all of the weights in the MLP network. So, overall, a set of input/desired output data values is used to train the entire network, the input set also realizing a corresponding set of network weights such that the error between the desired output signals and the actual network output signals is minimized in terms of the average overall learning points. The back propagation algorithm employs the steepest-descent method to arrive at a minimum of the mean squared error function. For one specific data pair, the error squared can be written as:

$$E_k = \frac{1}{2}\sum_{i=1}^{m}(y_{dk} - y_{ik})^2 = \frac{1}{2}\sum_{i=1}^{m} e_{ik}^2$$

where m neurons are assumed to be present, and y_{ik} is the ith neuron's kth output value.

The global error is then found by minimizing E_k over all the data set. If the number of data values that are present is N, then we have:

$$E = \sum_{k=1}^{N} E_k$$

This error function can then be minimized in batch mode or recursively in an on-line manner.

The network is now fully trained on the data presented and can be employed with any further data, although it may be desirable to present the data again cyclically until the overall error falls below a previously defined minimum value, i.e. until the weights converge. An important feature then emerges in that the MLP network has the ability to generalize when it is presented with new data not previously dealt with.

3-3 - Proposed Neural-Network Controller:

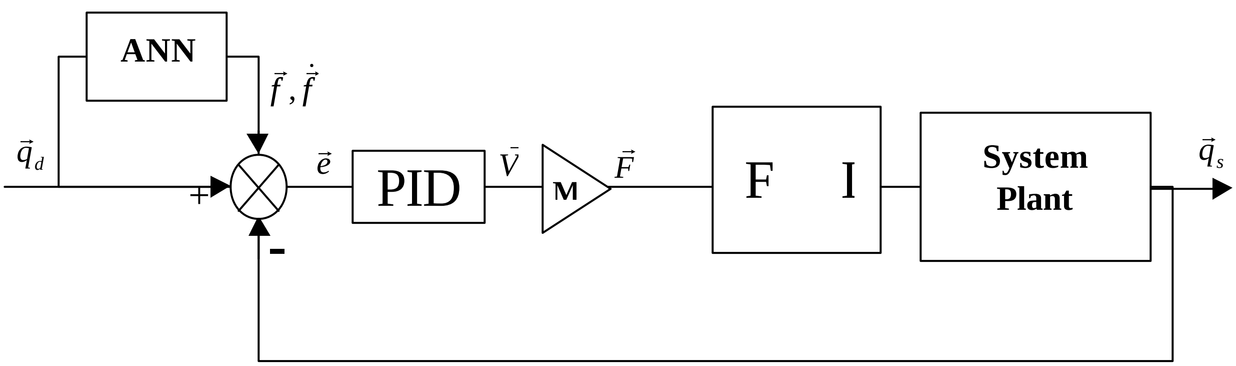
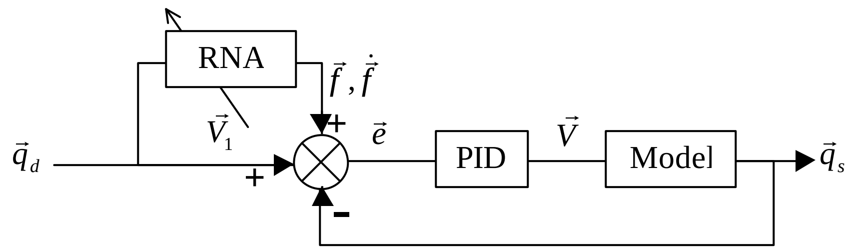


Fig. 7: Neural network control loop

Our goal is to find a method that will not modify the basic structure of the controlled process. So, with the standard control technique, we add an artificial neural network controller (fig 7) that anticipates the desired input of the closed loop system consisting of a PID controller in cascade with the system plant. In this structure, the neural network controller is used to modify the desired trajectory instead of generating a compensating torque. The five output positions are represented by vector $\vec{q}_s$, and the desired positions by vector $\vec{q}_d$. The vector $\vec{f}$ represents the output of the neural network controller. This control loop will be condensed and is represented by figure 8:



<u>Fig. 8:</u> Condensed neural network control loop

The vector $\vec{V}$ represents the output of the PID regulator.
The vector $\vec{e}$ represents the error after the insertion of the ANN controller.
The vector $\vec{e}$ represents the error before insertion of the ANN controller.
So, the error vector $\vec{e}$ between the actual output vector $\vec{q}_s$ and the desired input vector $\vec{q}_d$ will be written as:

$$\vec{e} = \vec{q}_d - \vec{q}_s$$

The vector $\vec{e}$ controlling the PID bloc and its derivative can be written as follows:

$$\vec{e} = \vec{q}_d - \vec{q}_s + \vec{f} = \vec{e} + \vec{f}$$
$$\dot{\vec{e}} = \dot{\vec{q}}_d - \dot{\vec{q}}_s + \dot{\vec{f}} = \dot{\vec{e}} + \dot{\vec{f}}$$

The output vector $\vec{V}$ of the PID bloc controlling the plant model can be written as follows:

$$\vec{V} = kp \cdot \vec{e} + kv.\dot{\vec{e}} + ki \cdot \int \vec{e}dt = kp \cdot \vec{\varepsilon} + kp \cdot \vec{\phi} + kv \cdot \dot{\vec{\varepsilon}} + kv \cdot \dot{\vec{\phi}} + ki \cdot \int \vec{\varepsilon}dt + ki \cdot \int \vec{\phi}dt$$

This equation can be written in the following manner:

$$\boxed{\vec{V} - kp\vec{f} - kv\dot{\vec{f}} - ki\int \vec{f}\,dt = kp\vec{e} + kv\dot{\vec{e}} + ki\int \vec{e}dt = \vec{V}_1}$$

So, the main objective is to minimize the vector $\vec{V}_1$ that yields in minimizing the vector $\vec{e}$. This procedure will yield the output vector to track the desired input vector.

<u>Remark</u>:
If we consider that the model is characterized by a function f connecting the input and the output vectors, we can write:

$$\vec{q}_S = f(\vec{V})$$

or

$$\vec{V} = f^{-1}(\vec{q}_S)$$

In term of this consideration, the equation describing the vector $\vec{V}_1$ will be written as follows:

$$\boxed{\vec{V}_1 = f^{-1}(\vec{q}_s) - kp\vec{f} - kv\dot{\vec{f}} - ki\int \vec{f}\,dt = kp\vec{e} + kv\dot{\vec{e}} + ki\int \vec{e}dt}$$

From this equation we recognize that the artificial neural network used in the closed loop system has to identify f^{-1} indirectly. (When $\vec{V}_1$ tends to zero)

Back-propagation algorithm:

At iteration n, the error to minimize is:

$$x(n) = \frac{1}{2}\vec{V}_1^{\,T}(n)\cdot\vec{V}_1(n)$$

with

$$\boxed{\vec{V}_1 = \vec{V} - kp\vec{f} - kv\dot{\vec{f}} - ki\int \vec{f}\,dt = kp\vec{e} + kv\dot{\vec{e}} + ki\int \vec{e}\,dt}$$

For neuron j of the output layer, this equation can be written as follows:

$$V_{1j}(n) = V_j(n) - kpf_j(n) - kv\dot{f}_j(n) - ki\int f_j(n)dt$$

Where:

$$x(n) = \frac{1}{2}\vec{V}_1^{\,T}(n)\cdot\vec{V}_1(n) = \frac{1}{2}\sum_{j\in C} V_{1j}^2(n)$$

C being the set of the neurons of the output layer.
And the average error will be equal to:

$$\xi_{av}(n) = \frac{1}{N}\sum_{n=1}^{N}\xi(n)$$

5 - NUMERICAL SIMULATION:

A number of neural network controllers have been trained using the back-propagation technique based on the first order algorithm with different learning rates and different momentum terms. Figure 9 shows an optimum learning rate equal to 0.08 and an optimum momentum term equal to 0.1. The activation function of the neuron is a sigmoide that uses the hyperbolic tangent with its parameter equal to 10^7. Figure 10 shows that the (5-9-5) multi-layer perceptron represents the optimal architecture with minimum average error equal to 0.1032.

Figures 11,12 and 13 show the desired input x(t), y_1(t), y_2(t), z_1(t), z_2(t) (red line) corresponding to the five axes of control of the AMB, and the output responses (blue line) obtained without using the neural network controller.

Figures 14,15 and 16 show the desired input x(t), y_1(t), y_2(t), z_1(t), z_2(t) (red line) corresponding to the five axes of control of the AMB, and the output responses (blue line) obtained during the learning process of the closed loop control combined with the neural network controller. We notice the complete accuracy of the output variables in following the desired input variables during the learning process.

Figures 17, 18 and 19 show the desired input x(t), y_1(t), y_2(t), z_1(t), z_2(t) (red line) corresponding to the five axes of control of the AMB, and the output responses (blue line) obtained after the learning process of the closed loop control combined with the neural network controller. We notice that the output variables of the system have changed from the ones obtained during the learning process, but they have better performances in response time and in precision compared to the control through the non-linear loop.

6 - COMMENTS AND PERSPCTIVES:

Applications in new technologies such as robotics, manufacturing, space technology, and medical instrumentation, as well as those in older technologies such as process control and aircraft control, are creating a wide spectrum of control problems in which non-linearity, uncertainties, and complexity play a major role. For the solution of many of these problems techniques based on ANN's are beginning to complement conventional control techniques [18], and in some cases they are emerging as the only viable alternatives. From a control theoretic point of view, ANN's may be considered as tractable parameterized families of nonlinear maps. As such they have found wide application in pattern recognition problems, which require nonlinear decision surfaces. With the introduction of dynamics and feedback, the scope of such networks as identifiers and controllers in nonlinear dynamical systems has increased significantly.

Our works aim to study the application of artificial neural networks (multi-layer perceptrons, MLPs) for the control of a Magnetic Levitation System by using the dynamic back-propagation method for the adjustment of parameters. The theoretical study of all the system and the simulated model are terminated. We finished the implementation of the different algorithms. We finished the simulation of the system in the linearized technique and the implementation of a neural network controller associated with a PID controller without making major modifications in the control of the process.

The results obtained were compared. Good results are obtained because this technique demonstrates the feasibility of controlling the AMB by using a neural network controller. Also the performances in precision and in response time are very important: it eliminates the over-shoot from the x(t) and diminish the response time of the five position variables.

A difference appears between the response of the system during the learning process and the final implementation of the algorithm. While complete concordance exits during the learning process, a different response appears after the learning process. This difference is due in particular to the number of trials that we have used during the learning process and to the technique used in the convergence of the neural network algorithms. We have used a first order algorithm which is the descent gradient convergence method in updating the weights of the network.

Our target from this work will be the real-time process (implementation on DSP of the neural networks controller and studying the behavior of all the process).

7 - REFERENCES:

[1]: W. T. Miller, R. S. Sutton, and P. J. Werbos (eds.), Neural networks for control, MIT Press, Cambridge, MA (1990).

[2]: K. Warwick, G. W. Irwin, and K. J. Hunt (eds.), Neural networks for control and systems, Peter Peregrinus Ltd (1992).

[3]: K. Warwick, Neural networks for control: counter arguments, Proc. IEE Int. Conference Control 94, Warwick University, pp.95-99 (1994).

[4]: K. J. Hunt, D. Sbarbaro, R. Zbikowski and P. J. Gawthrop, Neural networks for control systems - a survey, Automatica, **28**, pp.1083-1112 (1992).

[5]: A. H. Slocum, Precision Machine Design. Englewood Cliffs, NJ: Prentice-Hall, 1992, pp. 625-639.

[6]: B. Shafai, S. Beale, P. LaRocca, and E. Cusson, "Magnetic bearing control systems and adaptive forced balancing," IEEE Contr. Syst. Mag., pp. 4-13, Apr.1994.

[7]: B. Jackson and Y. Destombes, "The potential application of the active magnetic bearings to high power electrical machines," Lse Eng.Bull., vol. 15, no. 2, pp. 5-9, 1985.

[8]: J. Salm and G. Schweitzer, "Modeling and control of a flexible rotor with magnetic bearings," in Proc. 3rd Int. Conf. Vibrations in Rotating Machinery, Univ. York, U.K., Sept. 11-13, 1984, pp. C277-C284.

[9]: F. Matsumura and T. Yoshimoto, "System modeling and control design of horizontal shaft magnetic bearing system," IEEE Trans. Magn., vol. Mag-22, May 1986.

[10]: F. Matsumura, K. Nakagawa, and M. Kido, "Magnetic bearings - A novel type of magnetic suspension," in Conf. Vibration Rotating Machinery, Institute of Mechanical Engineering, Cambridge, U.K., Sept. 1986.

[11]: K. Youcef-Toumi and S. Reddy," Dynamics analysis and control of high speed and high precision active magnetic bearings," ASME J. Dynamic Syst., Measurement, Contr., vol. 114, Dec. 1992.

[12]: M. Najjar and C. Nasr, Non Linear Control of an Active Magnetic Bearing, DEA memory, Lebanese University, September 2000.

[13]: K.S.Narendra and K. Parthasarathy, Identification and control of dynamical systems using neural networks, IEEE Trans. on Neural Networks, **1**, pp.4-27 (1990).

[14]: D.H.Ballar, Cortical connections and parallel processing: structure and function. In Vision, brain and cooperative computation, M.Arbib and Hamson (eds.), pp.563-621, MIT Press, Cambridge, MA (1988).

[15]: D.E.Rumelhart and J.L.McClelland (eds.), Parallel distributed processing: explorations in microstructure of cognition, **1**: Foundations, MIT Press, Cambridge, MA (1986).

[16]: F.J.Pineda, Recurrent back propagation and dynamical approach to adaptive neural computation, Neural computation, **1**, pp.162-172 (1989).

[17]: K.S.Narendra and K.Parthasarathy, Gradient methods for the optimization of dynamical systems containing neural networks, IEEE Trans. on Neural Networks, **2**, pp.252-262 (1991).

[18]: K.S.Narendra, Neural Networks for Control: Theory and Practice, Proceedings of the IEEE, vol.84 n°10, (1996).

****

This work was funded by the U. S. Air Force Office of Scientific Research (AFOSR), the European Office of Aerospace Research and Development (EOARD) under the supervision of Dr. Charbel N Raffoul.

****

RESULTS OF THE SIMULATION

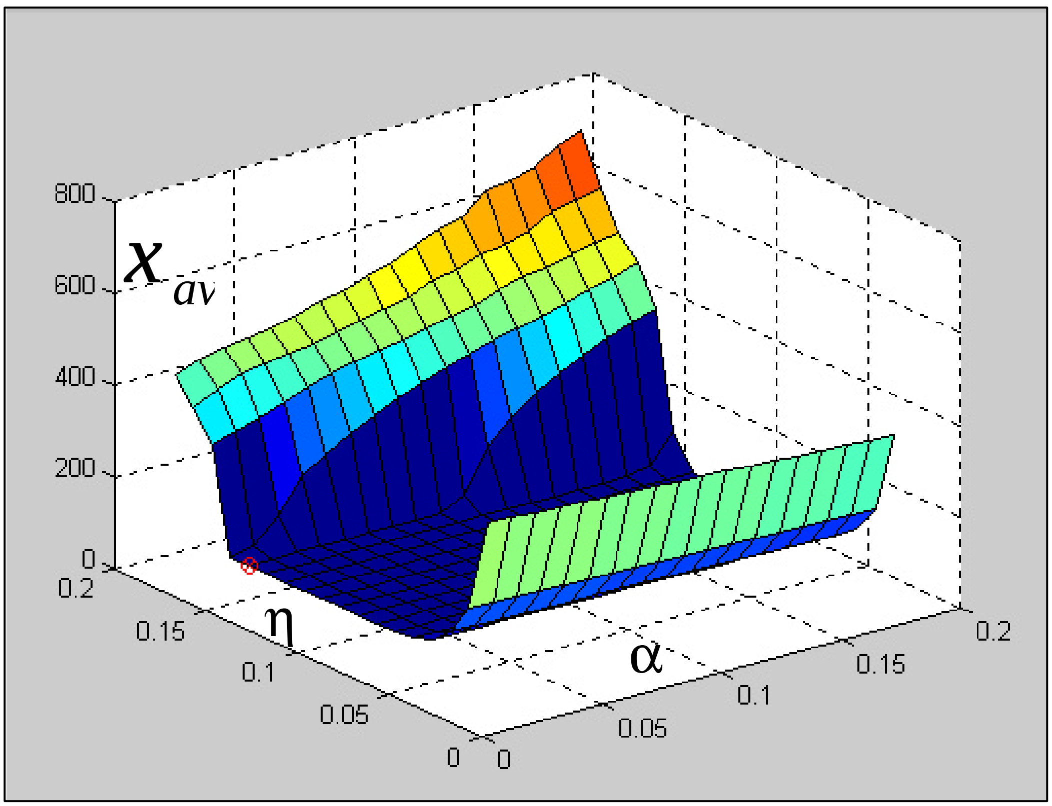


Fig. 9: Average error x_{av} function of the parameters α and η

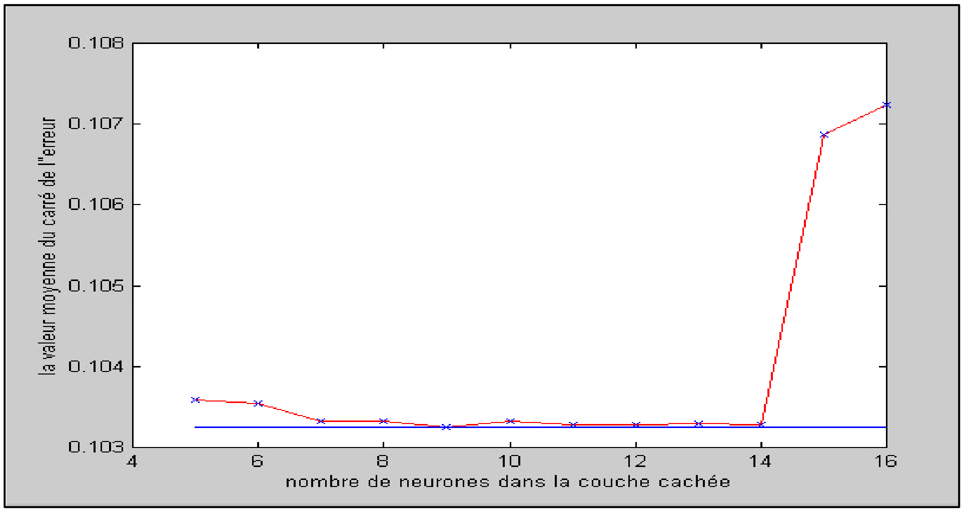


Fig.10: Average error x_{av} times the number of the hidden neurons in the MLP

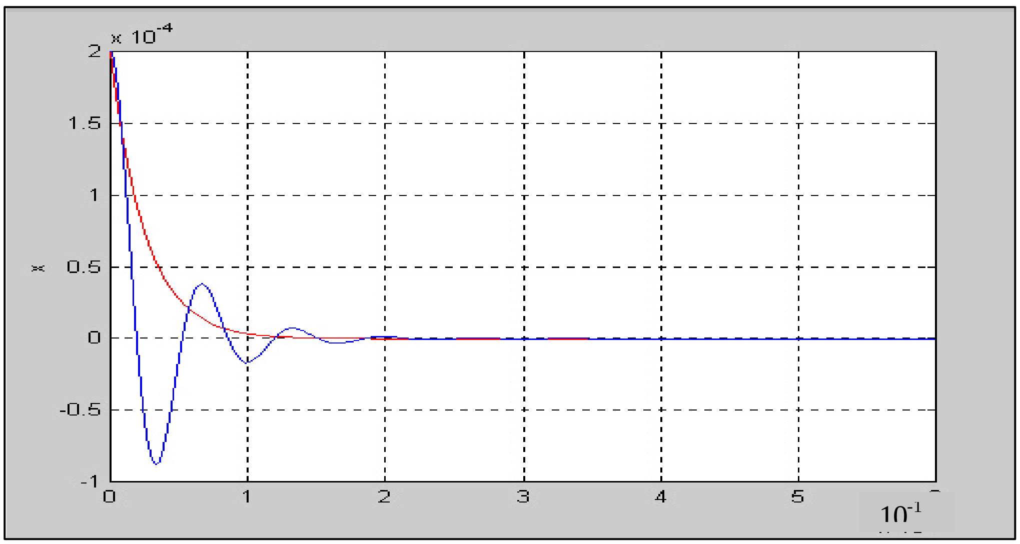


Fig. 11: Desired input x(t) (red line) and output response (blue line) without NN controller

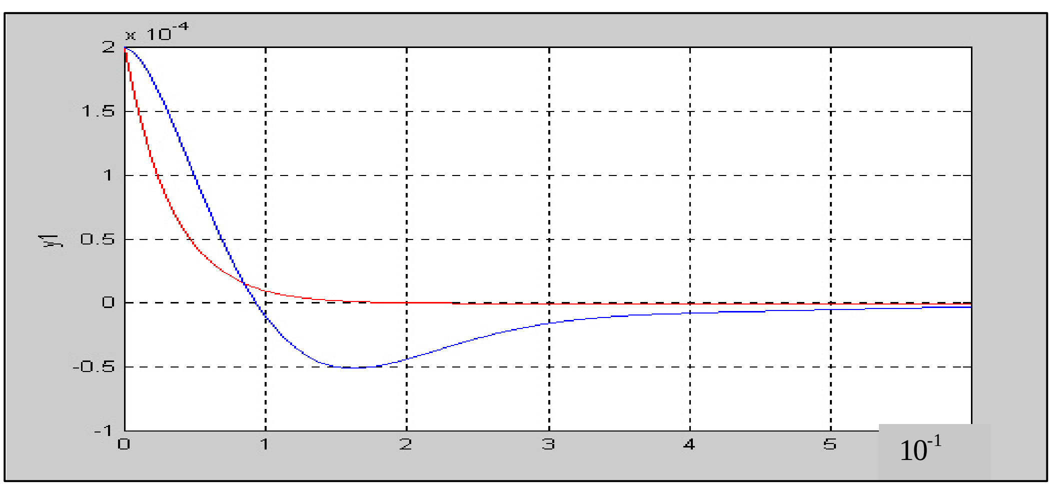


Fig. 12: Desired input y_1(t) (red line) and output response (blue line) without NN controller

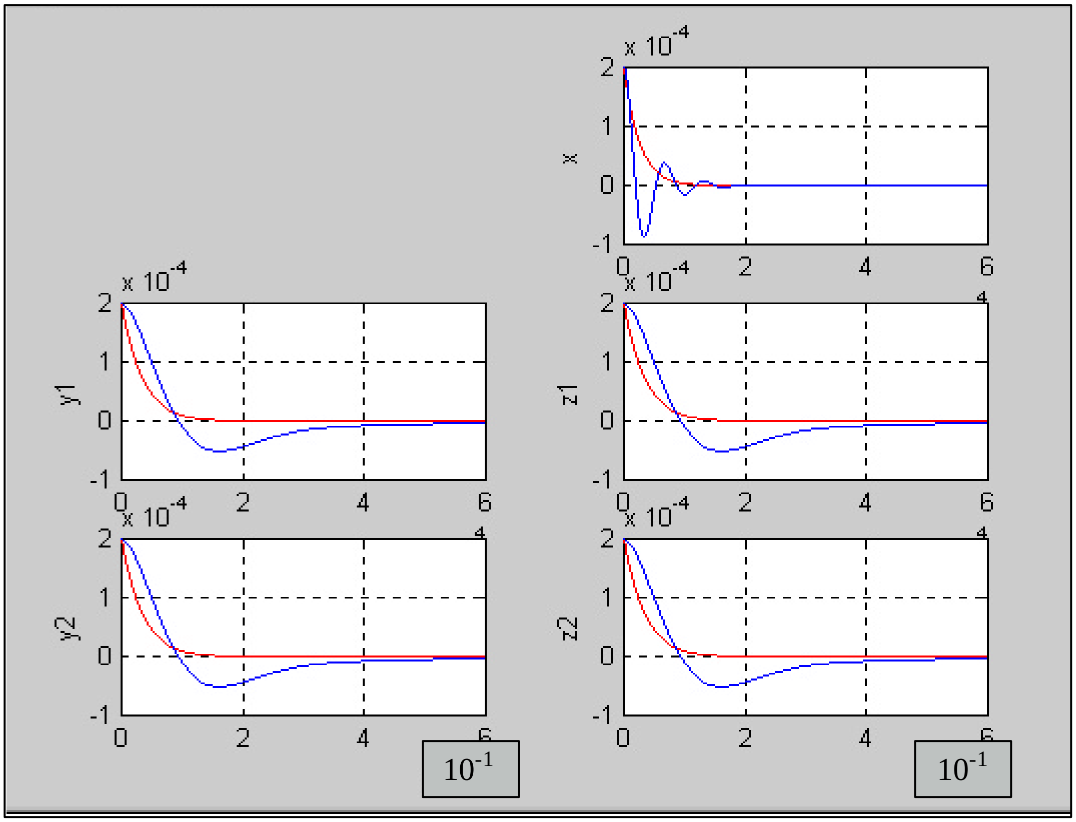


Fig. 13: Desired input x(t), y_1(t), y_2(t), z_1(t), z_2(t) (red line) and output responses (blue line) without NN controller.

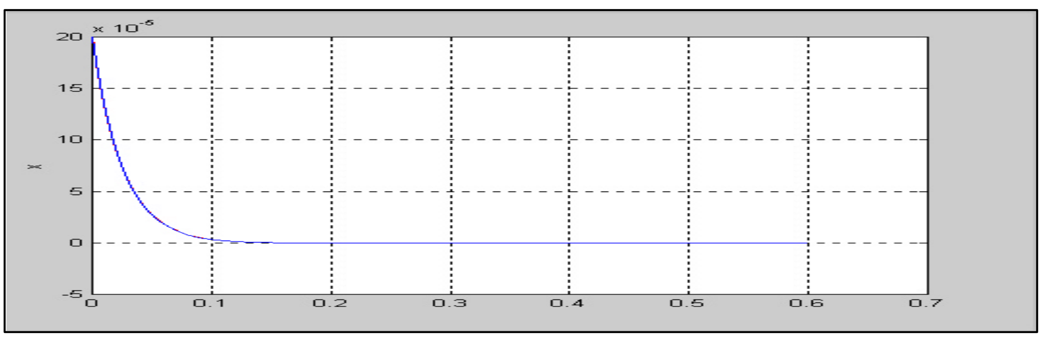


Fig. 14: Desired input x(t) (red line) and output response (blue line) during the learning process with NN controller.

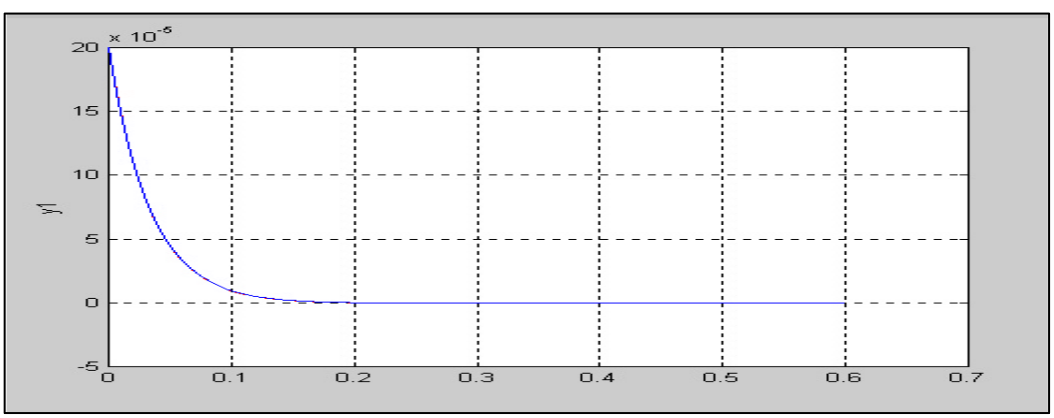


Fig. 15: Desired input y_1(t) (red line) and output response (blue line) during the learning process with NN controller (off-line).

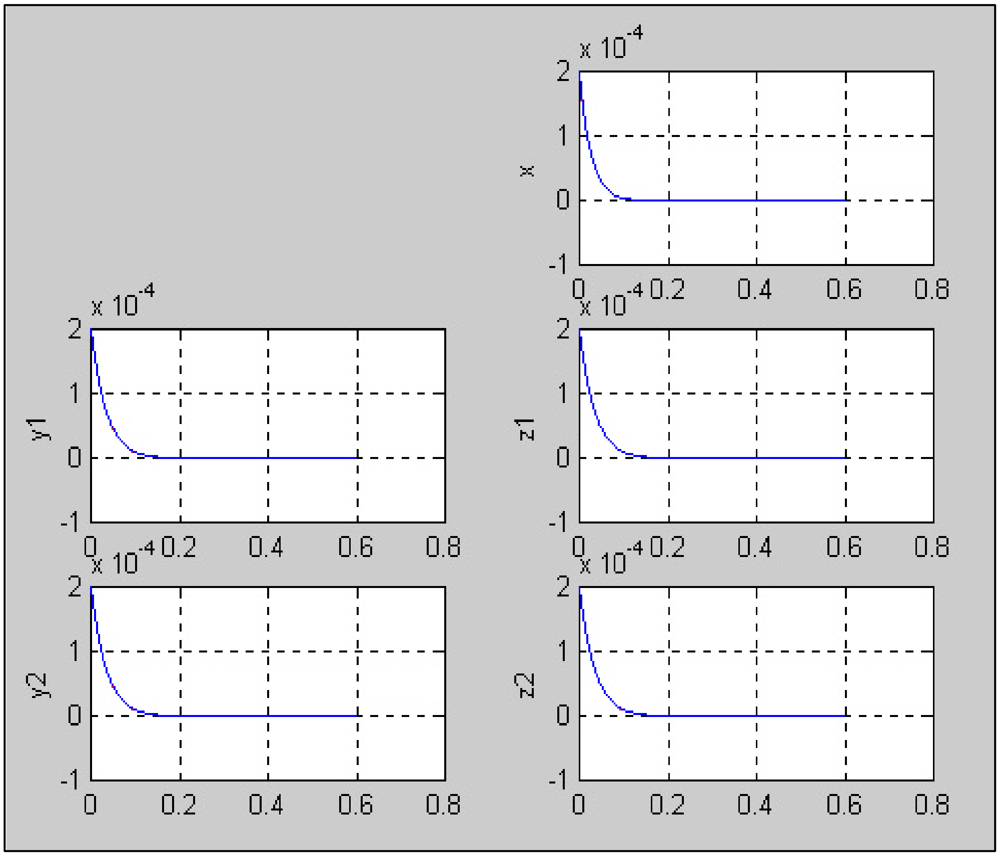


Fig. 16: Desired input x(t), y_1(t), y_2(t), z_1(t), z_2(t) (red line) and output responses (blue line) during the learning process with NN controller (off-line)

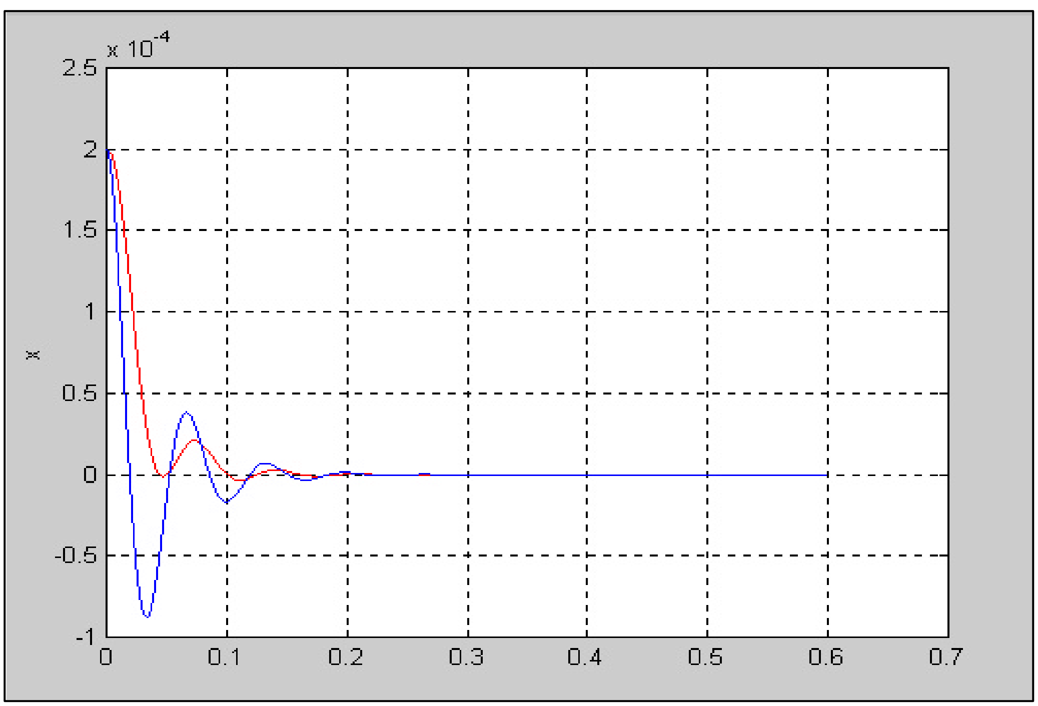


Fig. 17: Output x(t) (red line) with NN controller (on-line) and output x(t) (blue line) without NN controller

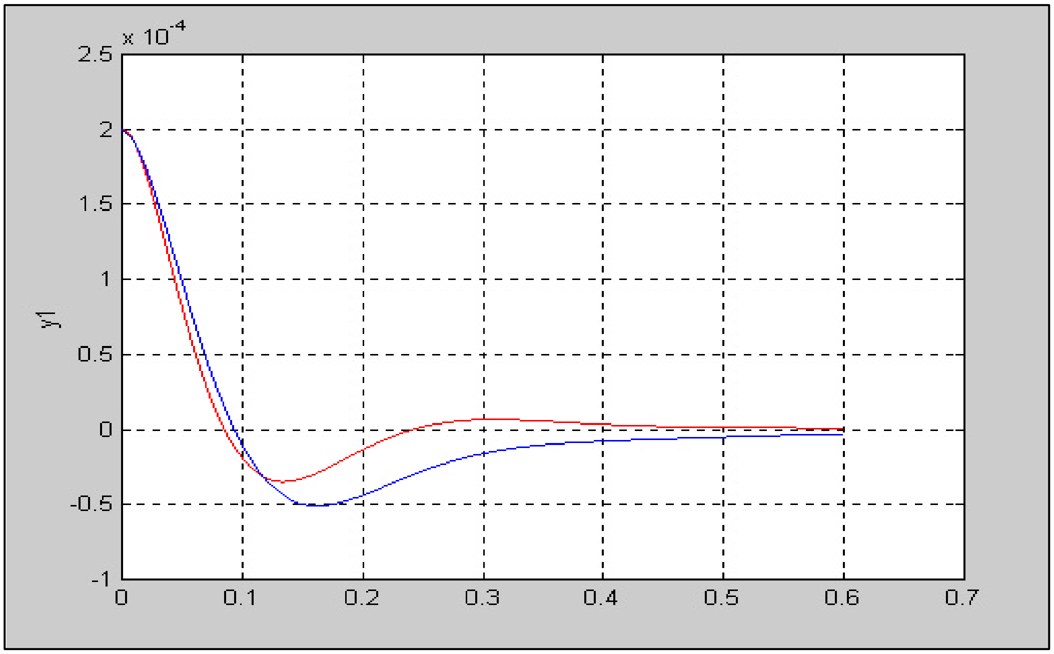


Fig. 18: Output $y_1(t)$ (red line) with NN controller (on-line) and output $y_1(t)$ (blue line) without NN controller

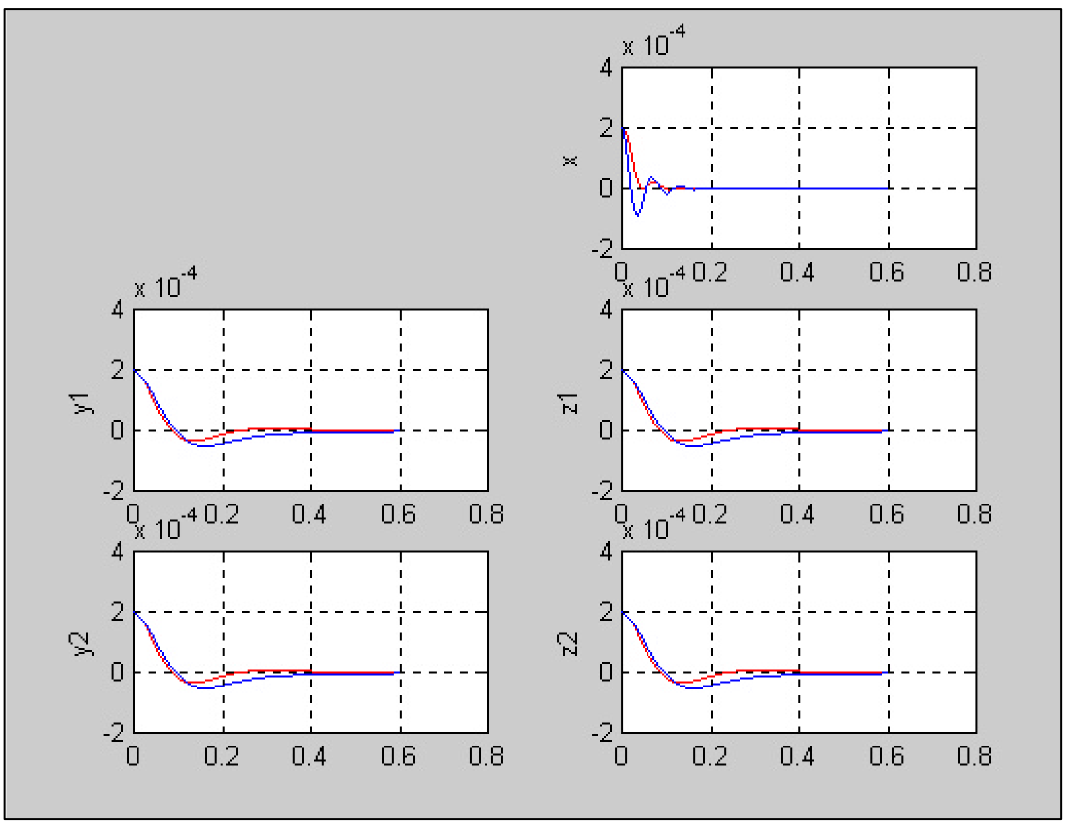


Fig. 19: Output $x(t)$, $y_1(t)$, $y_2(t)$, $z_1(t)$, $z_2(t)$ (red line) with NN controller (on-line) and output $x(t)$, $y_1(t)$, $y_2(t)$, $z_1(t)$, $z_2(t)$ (blue line) without NN controller